UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/726,338	12/01/2003	Kazi A. Zaman	OIC0185US	5478

60975 7590 08/18/2008
CAMPBELL STEPHENSON LLP
11401 CENTURY OAKS TERRACE
BLDG. H, SUITE 250
AUSTIN, TX 78758

EXAMINER	
PHAM, MICHAEL	

ART UNIT	PAPER NUMBER
2167	

MAIL DATE	DELIVERY MODE
08/18/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/726,338	ZAMAN ET AL.	
	Examiner	Art Unit	
	MICHAEL PHAM	2167	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 May 2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *3-22 and 31* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *3-22 and 31* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

Claim Status

1. Claims 3-22 and 31 are pending.

2. Claims 3-22 and 31 have been examined.

Claim Objections

3. Claim 31 is objected to because of the following informalities: claim 31 has a typo on line 7. Appropriate correction is required.

Claim Rejections - 35 USC § 101

4. Prior 101 rejections directed towards claims 3-18, 21-22, and 31 are withdrawn.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. Claims 3, 6-10, 14-22, and 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. patent 7366730 by Greenfield et. al. (hereafter Greenfield) further in view of U.S. Patent application publication 20020091681 by Cras et. al. (hereafter Cras).

<u>Claim 3 :</u>

Greenfield discloses the following claimed limitations:

"generating a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data source," [abstract, multidimensional data managed by a relational database system is facilitated by registration of solved cubes with a RDBMS. Col. 9 lines 9-10, OLAP schemas in RDBMS storage. Col. 12 lines 38-39, metadata that defines a cube is registered with the relational database management system. Accordingly, disclosing generating a relational model of a multidimensional data source (RDBMS) using one or more of a schema for the multidimensional data source (OLAP schemas) and metadata for the multidimensional data source (metadata)]

"wherein the relational model comprises a relational-to-multidimensional mapping between a virtual relational table <u>corresponding to the multidimensional data source</u> and the multidimensional data source, and the schema and metadata are accessed from the multidimensional data source;"[abstract, multidimensional data managed by a relational database system is facilitated by registration of solved cubes with a RDBMS. Col. 9 lines 10-12, use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. Col. 9 lines 45, virtual table. Figure 2 element 212. col. 12 lines 34-35 and 38-39, metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions. Accordingly, disclosing wherein the relational model (RDBMS) comprises a relational-to-multidimensional mapping (map) between a virtual relational table (figure 2 element 212) corresponding to the multidimensional data source (figure 2 element 210) and the multidimensional data source (figure 2 element 210), and the schema (schema) and metadata (metadata) are accessed from the multidimensional data

source (metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions)]

"forming the relational database query against the relational model of the multidimensional data source"[col. 13 lines 26-28, SQL query may be generated by or in conjunction with SQL generator of database server. col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly disclosing forming (generated) the relational database query (SQL) against the relational model of the multidimensional data source (RDBMS).]

"receiving the relational database query, the received relational database query being drawn against the relational model of the multidimensional data source;"[col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly, disclosing receiving the relational database query (SQL), the received relational database query being drawn against the relational model of the multidimensional data source(performing operations on multidimensional data managed by a RDBMS)]

"using the relational-to-multidimensional mapping together with relational/multidimensional equivalency logic to construct a multidimensional database query based on the received relational database query, wherein"[Col. 9 lines 10-15, one supporting method involves the use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. An OLAP API to relational database can recognize the views and hence the type and form of data returned. Thus, an OLAP API can generate an appropriate SQL statement for querying. Accordingly disclosing, using the relational-to-

multidimensional mapping (map) together with relational/multidimensional equivalency logic (figure 2 element 202) to construct a multidimensional database query (querying OLAP objects) based on the received relational database query (can generate SQL statement)]

"the relational/multidimensional equivalency logic comprises a general mapping between relational queries and structures and multidimensional queries and structures; and"[figure 2 element 202 and col. 9 lines 10-15 one supporting method involves the use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. An OLAP API to relational database can recognize the views and hence the type and form of data returned. Thus, an OLAP API can generate an appropriate SQL statement for querying. Accordingly, the relational/multidimensional equivalency logic (figure 2 element 202) comprises a general mapping (map) between relational queries (SQL) and structures (native relational schema) and multidimensional queries (SQL statement for querying OLAP objects) and structures (schema model)]

"submitting the constructed multidimensional database query for execution against the modeled multidimensional data source and"[col. 13 lines 26-28, SQL query may be generated by or in conjunction with SQL generator of database server. col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly, submitting the constructed multidimensional database query (generated SQL) for execution against the modeled multidimensional data source (rdbms)]

Greenfield discloses the virtual relational table corresponding to the multidimensional data source [figure 2 element 212]

Greenfield does not explicitly disclose:

"using a graphical user interface, wherein the graphical user interface

"displays a presentation layer representation of the " information "corresponding to the multidimensional data source, and"

"enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"

On the other hand, Cras discloses " using a graphical user interface, wherein the graphical interface [0075, manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.]

"displays a presentation layer representation of the "information "corresponding to the multidimensional data source, and"[0075, manipulation can be dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source)]

"enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"[abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data source). Abstract a query is generated automatically and deduced from a report definition (for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

"displaying a result of the constructed multidimensional database query against the modeled multidimensional data source"[0012, a system uses an inventive report then query metaphor, according to the present invention to create an analytical report on top of a multidimensional data model built on top of a relational or multidimensional database. 0074, Once the report has been defined the data to populate such a report is retrieved to build the document. Accordingly, Displaying a result (report) of the constructed multidimensional database query (query) against the modeled data source (multidimensional data model).]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Greenfield for the purpose of applying a better and efficient graphical user interface for the display of Greenfield.

Claim 6 :

Greenfield discloses:

"the relational query specifies a detail filter against the relational model, and wherein the constructed multidimensional query specifies that the detail filter be applied to the modeled multidimensional data source" [figure 3].

However, Greenfield does not explicitly disclose "the graphical user interface is further configured to enable selection of a detail filter to apply against the relational model".

On the other hand, Cras discloses, abstract, creating an analytic report on top of a multidimensional data model built on top of a relational or multidimensional database. The database operates in a computer system and provides returned values responsive to queries. Such a query is generated automatically and deduced from a report definition. 0075, disclosing that such manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.

Accordingly, Cras discloses "the graphical user interface is further configured to enable selection of a detail filter" [abstract, query is generated automatically and deduced from a report definition (selection of detail filter). 0075, dragging and dropping graphical objects]

Both Greenfield and Cras are directed to systems of multidimensional data, and are therefore both within the same field of endeavor. Cras discloses in more explicit detail the use of graphical objects in order to create a report and query information as described in detail above. For the above reasons it would have been obvious to one of an ordinary skill in the art to have applied

Cras's disclosure of dragging and dropping the graphical objects onto the report while the user is building the report and that a query is automatically generated and deduced from a report definition to the system of Greenfield in order to provide a more explicit means of communicating to the system the information the user requires. As stated in Cras, 0043, the GUI enables associated dimension objects to be filtered by selected member data so that report data is only replicated for selected members.

Claim 7 :

The combination of Greenfield and Cras discloses “wherein the relational query is expressed in SQL, and wherein the detail filter specified by the relational query is an SQL WHERE clause.” [col. 5 lines 36, sql where clause]

Claim 8 :

The combination of Greenfield and Cras disclose in Greenfield “wherein the relational query specifies an aggregation function against the relational model, and wherein the constructed multidimensional query specifies that the aggregation function be applied to the modeled multidimensional data source.”[col. 5 line 33. and col. 5 lines 27-28]

Claim 9 :

The combination of Greenfield and Cras disclose in Greenfield “wherein the relational query is expressed in SQL, and wherein the aggregation function specified by the relational query is an

SQL GROUP BY clause."[col. 5 lines 33-34]

Claim 10 :

Greenfield discloses "the relational query specifies a summary filter against the relational model, and wherein the constructed multidimensional query specifies the summary filter be applied to the modeled multidimensional data source" [figure 3]

However does not explicitly disclose "wherein the graphical user interface enables selection of a summary filter to apply against the relational model" On the other hand, Cras discloses, abstract, creating an analytic report on top of a multidimensional data model built on top of a relational or multidimensional database. The database operates in a computer System and provides returned values responsive to queries. Such a query is generated automatically and deduced from a report definition. 0075, disclosing that such manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.

Accordingly, Cras discloses "wherein the graphical user interface is further configured to enable selection of a summary filter to apply against the relational model" [abstract, query is generated automatically and deduced from a report definition (selection of a summary filter). 0075, dragging and dropping graphical objects]

Both Greenfield and Cras are directed to systems of multidimensional data, and are therefore' both within the same field of endeavor. Cras discloses in more explicit detail the use of graphical objects in order to create a report and query information as described in detail above. For the above reasons it would have been obvious to one of an ordinary skill in the art to have applied

Cras's disclosure of dragging and dropping the graphical objects onto the report while the user is building the report and that a query is automatically generated and deduced from a report definition to the system of Greenfield in order to provide a more explicit means of communicating to the system the information the user requires. As stated in Cras, 0043, the GUI enables associated dimension objects to be filtered by selected member data so that report data is only replicated for selected members.

Claim 14 :

The combination of Greenfield and Cras disclose in Greenfield “retrieving metadata identifying an aggregation function used for the selected measure of the multidimensional data source;” [col. 13 lines 7-9]

“determining whether the aggregation function identified by the metadata matches the selected aggregation function; and”[col. 13 lines 4-6]

“if the aggregation function identified by the metadata matches the selected aggregation function, generating a multidimensional query against the multidimensional data source that relies on the aggregation function performed in the multidimensional data source.”[col. 13 lines 26-39]

Claim 15 :

The combination of Greenfield and Cras disclose in Greenfield

“receiving, in response to submitting the multidimensional database query, a multidimensional database query result; and”[figure 3 element 308, fulfilled request]

"using a relational-to-multidimensional mapping contained by the model together with relational/multidimensional equivalency logic to construct a relational database query result based on the received multidimensional database query result."[figure 3 element 308]

Claim 16 :

The combination of Greenfield and Cras disclose in Greenfield

"determining that the received relational database query is drawn against both the relational model of the multidimensional data source and one or more native relational tables; and" [fig 2 element 212, col. 11 lines 58-64 and col. 12 lines 1-8, determining that the received relational database query is drawn against both the relational model of the multidimensional data source (rdbms) and one or more native relational tables (fig 2 element 212)]

"constructing a native relational database query based on aspects of the received relational database query drawn against conventional relational tables; and"[fig 2 element 212, col. 11 lines 58-64 and col. 12 lines 1-8, constructing a native relational database query (sql) based on aspects of the received relational database query drawn against conventional relational tables (212)]

"submitting the native relational database query for execution against the conventional relational tables, and wherein the constructed multidimensional database query is based on aspects of the received relational database query drawn against the relational model of the multidimensional data source, the method further comprising: receiving, in response to submitting the native relational database query,"[fig 2 element 212, col. 11 lines 58-64 and col. 12 lines 1-8, col. 13 lines 38-39, submitting the native relational database query (sql) for

execution against the conventional relational tables (fig 2 element 212), and wherein the constructed multidimensional database query (sql & procedures) is based on aspects of the received relational database query drawn against the relational model of the multidimensional data source (sql), the method further comprising: receiving, in response to submitting the native relational database query (sql & procedures)]

"a native relational database query result; and"[col. 13 lines 38-89, a native relational database query result (sql & procedure)]

"combining the constructed relational database query result with the received native relational database query result in accordance with the received relational database query."[col. 13 lines 38-89, combining the constructed relational database query result with the received native relational database query result in accordance with the received relational database query(SQL and procedures)]

Claim 17 :

The combination of Greenfield and Cras disclose in Greenfield "making information about the model available for use in building the received relational database query."[abstract]

Claim 18 :

The combination of Greenfield and Cras disclose in Greenfield

"determining that the received relational database query is drawn against both the relational model of the multidimensional data source and one or more native relational tables; and"[fig 2 element 212, col. 11 lines 58-64 and col. 12 lines 1-8, determining that the received

relational database query is drawn against both the relational model of the multidimensional data source (rdbms) and one or more native relational tables (fig 2 element 212)]

“constructing a native relational database query based on aspects of the received relational database query drawn against conventional relational tables; and” [fig 2 element 212, col. 11 lines 58-64 and col. 12 lines 1-8, constructing a native relational database query (sql) based on aspects of the received relational database query drawn against conventional relational tables (212)]

“submitting the native relational database query for execution against the conventional relational tables, and”[s col. 13 lines 38-89, submitting the native relational database query (SQL and procedure) for execution against the conventional relational tables (sql statement is executed against a table)]

“wherein the constructed multidimensional database query is based on aspects of the received relational database query drawn against the relational model of the multidimensional data source.”[col. 13 lines 38-89, wherein the constructed multidimensional database query (SQL statement and procedures) is based on aspects of the received relational database query (SQL) drawn against the relational model of the multidimensional data source (RDBMS)]

Claim 19:

Greenfield discloses the following claimed limitations:

"a first set of instructions, executable on a processor, configured to generate a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data source,“ [abstract, multidimensional data managed by a relational database system is facilitated by registration of solved cubes with a

RDBMS. Col. 9 lines 9-10, OLAP schemas in RDBMS storage. Col. 12 lines 38-39, metadata that defines a cube is registered with the relational database management system. Accordingly, disclosing a first set of instructions, executable on a processor, generating a relational model of a multidimensional data source (RDBMS) using one or more of a schema for the multidimensional data source (OLAP schemas) and metadata for the multidimensional data source (metadata)]

"wherein the relational model comprises a relational-to-multidimensional mapping between a virtual relational table corresponding to the multidimensional data source and the multidimensional data source, and" [abstract, multidimensional data managed by a relational database system is facilitated by registration of solved cubes with a RDBMS. Col. 9 lines 10-12, use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. Col. 9 lines 45, virtual table. Figure 2 element 212. col. 12 lines 34-35 and 38-39, metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions. Accordingly, disclosing wherein the relational model (RDBMS) comprises a relational-to-multidimensional mapping (map) between a virtual relational table (figure 2 element 212) corresponding to the multidimensional data source (figure 2 element 210) and the multidimensional data source (figure 2 element 210)]

"the schema and metadata are accessed from the multidimensional data source;"[abstract, multidimensional data managed by a relational database system is facilitated by registration of solved cubes with a RDBMS. Col. 9 lines 10-12, use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. Col. 9 lines 45, virtual table. Figure 2 element 212. col. 12 lines 34-35 and 38-39, metadata that

defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions. Accordingly, disclosing and the schema (schema) and metadata (metadata) are accessed from the multidimensional data source (metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions)]

"a second set of instructions, executable on a processor, configured to form the relational database query against the relational model of the multidimensional data source"[col. 13 lines 26-28, SQL query may be generated by or in conjunction with SQL generator of database server. col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly disclosing a second set of instructions, executable on a processor, configured to form (generated) the relational database query (SQL) against the relational model of the multidimensional data source (RDBMS).]

"a third set of instructions, executable on the processor, configured to receive the relational database query, the received relational database query being drawn against the relational model of the multidimensional data source;"[col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly, disclosing a third set of instructions, executable on the processor, configured to receive the relational database query (SQL), the received relational database query

being drawn against the relational model of the multidimensional data source (performing operations on multidimensional data managed by a RDBMS)]

“a fourth set of instructions, executable on the processor configured to use the relational-to-multidimensional mapping to translate the received relational database query into a multidimensional database query;”[Col. 9 lines 10-15, one supporting method involves the use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. An OLAP API to relational database can recognize the views and hence the type and form of data returned. Thus, an OLAP API can generate an appropriate SQL statement for querying OLAP objects. Accordingly, disclosing a fourth set of instructions, executable on the processor configured to use the relational-to-multidimensional mapping (map) to translate the received relational database query (generate SQL) into a multidimensional database query (querying OLAP objects)]

“a fifth set of instructions, executable on the processor, configured to submit the multidimensional database query for execution against the modeled multidimensional data source; and”[col. 13 lines 26-28, SQL query may be generated by or in conjunction with SQL generator of database server. col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly, a fifth set of instructions, executable on the processor, configured to submit the multidimensional database query (generated SQL and procedures) for execution against the modeled multidimensional data source (rdbms)]

Greenfield discloses the virtual relational table corresponding to the multidimensional data source [figure 2 element 212].

Green field does not explicitly disclose “using a graphical user interface, wherein the graphical user interface

displays a presentation layer representation of the” information " corresponding to the multidimensional data source on display”

“enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;”

“a sixth set of instructions, executable on the processor, configured to display a result of the multidimensional database query against the modeled multidimensional data source.”

On the other hand, Cras discloses “ using a graphical user interface, wherein the graphical interface [0075, manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.]

“displays a presentation layer representation of the “information ”corresponding to the multidimensional data source on the display, and”[0075, manipulation can be

dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source on display)]

"enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"[abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data source). Abstract a query is generated automatically and deduced from a report definition (for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

""a sixth set of instructions, executable on the processor, configured to display a result of the constructed multidimensional database query against the modeled multidimensional data source"[0012, a system uses an inventive report then query metaphor, according to the present invention to create an analytical report on top of a multidimensional data model built on top of a relational or multidimensional database. 0074, Once the report has been defined the data to populate such a report is retrieved to build the document. Accordingly, a sixth set of instructions, executable on the processor, configured to display a result (report) of the constructed multidimensional database query (query) against the modeled data source (multidimensional data model).]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Greenfield for the purpose of applying a better and efficient graphical user interface for the display of Greenfield.

Claim 20:

The combination of Greenfield and Cras disclose in Greenfield

“a seventh set of instructions, executable on the processor, configured to receive, in response to submitting the multidimensional database query, a multidimensional database query result; and"[col. 11 lines 29, a seventh set of instructions, executable on the processor, configured to receive, in response to submitting the multidimensional database query, a multidimensional database query result (col. 11 lines 29, results)]

“an eighth set of instructions, executable on the processor, configured to use a relational-to-multidimensional mapping to translate the received multidimensional query result into a relational database query"[Col. 9 lines 10-15, one supporting method involves the use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. An OLAP API to relational database can recognize the views and hence the type and form of data returned. Thus, an OLAP API can generate an appropriate SQL statement for querying OLAP objects. Accordingly, disclosing a eighth set of instructions, executable on the processor configured to use the relational-to-multidimensional mapping (map) to translate the

received relational database query (generate SQL) into a multidimensional database query (querying OLAP objects)]

Claim 21:

Greenfield discloses the following claimed limitations:

"a processor;"[figure 4]

"a display coupled to the processor;"[figure 4]

"a modeling subsystem configured to execute on the processor and further configured to generate a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data source, wherein the relational model comprises a relational-to-multidimensional mapping between the virtual relational table and the multidimensional data source, and"[See figure 4 and figure 2. See col. 9 lines 10-15, and abstract lines 1-3. Accordingly, disclosing a modeling subsystem configured to execute on the processor (figure 4) and further configured to generate a relational model of a multidimensional data source (RDBMS) using one or more of a schema for the multidimensional data source (OLAP schema) and metadata for the multidimensional data source (metadata), wherein the relational model comprises a relational-to-multidimensional mapping (map) between the virtual relational table (figure 2 element 212) and the multidimensional data source (figure 2 element 210),]

"the schema and metadata are accessed from the multidimensional data source;"[abstract, multidimensional data managed by a relational database system is facilitated by registration of

solved cubes with a RDBMS. Col. 9 lines 10-12, use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. Col. 9 lines 45, virtual table. Figure 2 element 212. col. 12 lines 34-35 and 38-39, metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions. Accordingly, disclosing the schema (schema) and metadata (metadata) are accessed from the multidimensional data source (metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions)]

“and further configured to form the relational database query against the relational model of the multidimensional data source” [col. 13 lines 26-28, SQL query may be generated by or in conjunction with SQL generator of database server. col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly disclosing a second set of instructions, executable on a processor, configured to form (generated) the relational database query (SQL) against the relational model of the multidimensional data source (RDBMS).]

“a query reception subsystem configured to execute on the processor and further configured to receive the relational database query, the received relational database query being drawn against the relational model of the multidimensional data source;” [col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly, disclosing a third set of instructions, executable on the processor, configured to receive the relational database query (SQL), the received relational

database query being drawn against the relational model of the multidimensional data source (performing operations on multidimensional data managed by a RDBMS)]

“a multidimensional query construction subsystem configured to execute on the processor and further configured to use the relational-to-multidimensional mapping to construct a multidimensional database query based on the received relational database query; and” [Col. 9 lines 10-15, one supporting method involves the use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. An OLAP API to relational database can recognize the views and hence the type and form of data returned. Thus, an OLAP API can generate an appropriate SQL statement for querying OLAP objects. Accordingly, disclosing a fourth set of instructions, executable on the processor configured to use the relational-to-multidimensional mapping (map) to translate (bridge) the received relational database query (generate SQL) into a multidimensional database query (querying OLAP objects)]

“a query submission subsystem configured to execute on the processor and further configured to submit the constructed multidimensional database query for execution against the modeled multidimensional data source." [col. 13 lines 26-28, SQL query may be generated by or in conjunction with SQL generator of database server. col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly, a fifth set of instructions, executable on the processor,

configured to submit the multidimensional database query (generated SQL and procedures) for execution against the modeled multidimensional data source (rdbms)]

Greenfield discloses the virtual relational table corresponding to the multidimensional data source [figure 2 element 212].

Greenfield does not explicitly disclose:

“a graphical user interface subsystem configured to execute on the processor” “wherein the graphical user interface subsystem further

Displays a presentation layer representation of” information “corresponding to the multidimensional data source on the display,

Enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;”

On the other hand, Cras discloses “ a graphical user interface subsystem configured on the processor” “wherein the graphical user interface subsystem further" [0075, manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.]

"displays a presentation layer representation of the "information "corresponding to the multidimensional data source on the display, and"[0075, manipulation can be dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source on the display)]

"enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"[abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data source). Abstract a query is generated automatically and deduced from a report definition (for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Greenfield for the purpose of applying a better and efficient graphical user interface for the display of Greenfield.

Claim 22:

The combination of Greenfield and Cras disclose in Greenfield "a query result reception subsystem that receives, in response to submitting the multidimensional database query, a

multidimensional database query result; and"[col. 11 lines 29, a query result reception subsystem that receives, in response to submitting the multidimensional database query (sql and procedure), a multidimensional database query result (col. 11 lines 29, results)]

"a relational query result construction subsystem that uses a relational-to-multidimensional mapping to construct a relational database query result based on the received multidimensional database query result."[Col. 9 lines 10-15, one supporting method involves the use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. An OLAP API to relational database can recognize the views and hence the type and form of data returned. Thus, an OLAP API can generate an appropriate SQL statement for querying OLAP objects. Accordingly, disclosing a relational query result construction subsystem that uses a relational-to-multidimensional mapping (map) to translate (bridge) the received relational database query (generate SQL) into a multidimensional database query (querying OLAP objects)]

Claim 31:

Greenfield discloses the following claimed limitations:

"generating a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data source," [See figure 4 and figure 2. See col. 9 lines 10-15, and abstract lines 1-3. Accordingly, disclosing a modeling subsystem configured to execute on the processor (figure 4) and further configured to generate a relational model of a multidimensional data source (RDBMS) using one or more of a

schema for the multidimensional data source (OLAP schema) and metadata for the multidimensional data source (metadata), wherein the relational model comprises a relational-to-multidimensional mapping (map) between the virtual relational table (figure 2 element 212) and the multidimensional data source (figure 2 element 210),]

“wherein the relational model comprises a relational-to-multidimensional mapping between a virtual relational table corresponding ot the multidimensional data source and the multidimensional data source, and"[See figure 4 and figure 2. See col. 9 lines 10-15, and abstract lines 1-3. Accordingly, disclosing wherein the relational model comprises a relational-to-multidimensional mapping (map) between a virtual relational table (212) corresponding ot the multidimensional data source and the multidimensional data source (210).]

“the schema and metadata are accessed from the multidimensional data source;” [abstract, multidimensional data managed by a relational database system is facilitated by registration of solved cubes with a RDBMS. Col. 9 lines 10-12, use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. Col. 9 lines 45, virtual table. Figure 2 element 212. col. 12 lines 34-35 and 38-39, metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions. Accordingly, disclosing the schema (schema) and metadata (metadata) are accessed from the multidimensional data source (metadata that defines a cube is registered; the multidimensional data is stored in a relational database and includes a plurality of dimensions)]

"forming the relational database query against a relational model of a multidimensional data source" [col. 13 lines 26-28, SQL query may be generated by or in conjunction with SQL generator of database server. col. 13 lines 35-39, SQL statement is executed against a table. Col. 13 lines 46-48, performing operations on multidimensional data managed by a RDBMS. Accordingly disclosing a second set of instructions, executable on a processor, configured to form (generated) the relational database query (SQL) against the relational model of the multidimensional data source (RDBMS).]

"receiving the relational database query, the received relational database query being drawn against both the relational model of a multidimensional data source and a native relational table;"[Accordingly disclosing receiving the relational database query (SQL), the received relational database query being drawn against both the relational model of a multidimensional data source (RDBMS) and a native relational table (figure 2 element 212)]

"converting the received relational database query into (1) a native relational database query against only the native relational table, and (2) a multidimensional database query against the multidimensional data source;" [col. 9 lines 14-15, generate an appropriate SQL statement for querying OLAP objects. col. 13 lines 25-34 and 35-39. converting (interpret and determines procedures) the received relational database query (SQL) into (1) a native relational database query (SQL) against only the native relational table (table), and (2) a multidimensional database query (multidimensional query) against the multidimensional data source (OLAP object)]

“submitting the native relational database query against the native relational table;"[col. 13 lines 35-40, submitting the native relational database query (SQL) against the native relational table (table)]

“submitting the multidimensional database query against the multidimensional data source;”[Col. 9 line 15, SQL statement for querying for OLAP objects. Col. 13 lines 37-39, base data is aggregated using a SQL statement and procedures are executed that manipulate the aggregated data. Accordingly, submitting the multidimensional database query (query) against the multidimensional data source (OLAP object)]

Greenfield does not explicitly disclose
“using a graphical user interface, wherein the graphical user interface”

“displays a presentation layer representation of the virtual relational table corresponding to the multidimensional data source, and”

“enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;”

“combining contents of a first search result produced in response to the native relational database query and a second search result produced in response to the multidimensional database query into a third search result responsive to the received relational database query; and”

“displaying the third search result.”

On the other hand, Cras discloses

“using a graphical user interface, wherein the graphical user interface”[0075, gui]

“displays a presentation layer representation of the” information “corresponding to the multidimensional data source, and”[[0075, manipulation can be dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source)]]

“enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;” [abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data source). Abstract a query is generated automatically and deduced from a report (and represented by the displayed presentation layer) definition (for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

“combining contents of a first search result produced in response to the native relational database query and a second search result produced in response to the multidimensional database query into a third search result responsive to the received relational database query; and”[0089, report engine creates a datasource query consisting of a tree of blockqueries and provides the dsquery to the cube. 0091, a datasource query is used to access and retrieve data from the OLAP and RDBMS data sources. 0104-0105. combining contents (report) of a first search result produced in response to the native relational database query (relational query) and a second search result produced in response to the multidimensional database query (multidimensional query) into a third search result (report) responsive to the received relational database query (relational query)]

“displaying the third search result.”[0132, datasource query contains the information necessary to build a query of the report, then query process.report]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Greenfield for the purpose of applying a better and efficient graphical user interface for the display of Greenfield.

7. Claims 4-5 and 12-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. patent 7366730 by Greenfield et. al. (hereafter Greenfield) and U.S. Patent application publication 20020091681 by Cras et. al. (hereafter Cras) further in view of U.S. patent application publication 20040039736 by Kilmer et. al. (hereafter Kilmer).

Claim 4 :

Greenfield and Cras do not explicitly disclose "wherein the multidimensional query is constructed in MDX."

However, Kilmer discloses 0019, "multi-dimensional expression query statement mdx". It would have been obvious to one of an ordinary skill in the art to apply KIimer's disclosure of MDX to the combination of Greenfield and Cras for the purpose of being able to talk to the multidimensional data source in a language it would understand. It is further well known in the art that MDX is a standard multidimensional database query language.

Claim 5 :

The combination of Greenfield, Cras, and Kilmer disclose in Greenfield “wherein the relational query is expressed in SQL.” [col. 13 lines 26-27, sql]

8. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. patent 7366730 by Greenfield et. al. (hereafter Greenfield) and U.S. Patent application publication 20020091681 by Cras et. al. (hereafter Cras) further in view of “Database Systems: the complete book” by Hector Garcia-Molina et. al. (hereafter Garcia-Molina).

Claim 11 :

The combination of Greenfield and Cras discloses in Greenfield "wherein the relational query is expressed in SQL" [col. 5 lines 24-25], "and wherein the summary filter specified by the relational query" [figure 3 element 304].

However, Greenfield does not explicitly disclose is an "SQL HAVING clause".

On the other hand, Garcia-Molina discloses that SQL supports grouping, by using a GROUP BY

clause [pages 277, 282]. All references are in the same field of endeavor, databases. It would have been obvious to have applied Garcia-Molina's disclosure above to the combination of Greenfield and Cras for the purpose of grouping tuples together. One of ordinary skill in the art would know this would be a form of filtering.

9. Claims 3, 6-10, 14-22, and 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. patent 5675785 by Hall et. al. (hereafter Hall) further in view of U.S. Patent application publication 20020091681 by Cras et. al. (hereafter Cras).

Claim 3:

Hall discloses the following claimed limitations:

"generating a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data source," [abstract lines 1-2, arranged in data tables, e.g. fact/reference tables. Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Accordingly, generating a relational model of a multidimensional data source (fact/reference table) using one or more of a schema for the multidimensional data source (schema) and metadata for the multidimensional data source (figure 2)]

"wherein the relational model comprises a relational-to-multidimensional mapping between a virtual relational table corresponding to the multidimensional data source and the multidimensional data source, and the schema and metadata are accessed from the multidimensional data source;"[Col. 6 lines 14-17, schema which abstracts the definition of the

physical data warehouse into one which is independent of which summaries are populated. Figure 2. Col. 6 lines 62-66, builds SQL based on this schema. Accordingly, wherein the relational model (fact/reference table) comprises a relational-to-multidimensional mapping (schema) between a virtual relational table (virtual table) corresponding to the multidimensional data source (data warehouse) and the multidimensional data source (data warehouse), and the schema (schema) and metadata (figure 2) are accessed from the multidimensional data source (data warehouse)]

"forming the relational database query against the relational model of the multidimensional data source"[col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. Accordingly, forming the relational database query (SQL) against the relational model of the multidimensional data source (fact/reference tables)]

"receiving the relational database query, the received relational database query being drawn against the relational model of the multidimensional data source;"[col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. Accordingly, receiving the relational database query (SQL), the received relational database query being drawn against the relational model of the multidimensional data source (fact/reference tables)]

"using the relational-to-multidimensional mapping together with relational/multidimensional equivalency logic to construct a multidimensional database query based on the received relational database query, wherein"[Col. 6 lines 62-66, builds SQL based on this schema. Col. 9 lines 61-64, converts the query from one which references virtual tables and columns into one which matches a physical instance of data in the data warehouse. Accordingly, using the relational-to-multidimensional mapping (schema) together with

relational/multidimensional equivalency logic (matches physical instance) to construct a multidimensional database query (convert query) based on the received relational database query (SQL)]

"the relational/multidimensional equivalency logic comprises a general mapping between relational queries and structures and multidimensional queries and structures; and"[Col. 6 lines 62-66, builds SQL based on this schema. Col. 9 lines 61-64, converts the query from one which references virtual tables and columns into one which matches a physical instance of data in the data warehouse. Accordingly, the relational/multidimensional equivalency logic (matches) comprises a general mapping between relational queries (SQL) and structures (virtual tables/columns) and multidimensional queries (converted queries) and structures (fact/reference table/ physical instance of data in the data warehouse)]

"submitting the constructed multidimensional database query for execution against the modeled multidimensional data source and"[abstract, modified query to obtain the particular information from the database warehouse. Col. 7 lines 2-4, converts the query to utilize the most appropriate fact and reference tables and issues the modified query to database server. Accordingly, submitting the constructed multidimensional database query (modified query) for execution against the modeled multidimensional data source (fact/reference table)]

Hall discloses the virtual relational table corresponding to the multidimensional data source [col. 3 lines 58-60, virtual table]

Hall does not explicitly disclose:

"using a graphical user interface, wherein the graphical user interface

"displays a presentation layer representation of the " information "corresponding to the multidimensional data source, and"

"enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"

On the other hand, Cras discloses " using a graphical user interface, wherein the graphical interface [0075, manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.]

"displays a presentation layer representation of the "information "corresponding to the multidimensional data source, and"[0075, manipulation can be dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source)]

"enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"[abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data source). Abstract a query is generated automatically and deduced from a report definition

(for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

"displaying a result of the constructed multidimensional database query against the modeled multidimensional data source"[0012, a system uses an inventive report then query metaphor, according to the present invention to create an analytical report on top of a multidimensional data model built on top of a relational or multidimensional database. 0074, Once the report has been defined the data to populate such a report is retrieved to build the document. Accordingly, Displaying a result (report) of the constructed multidimensional database query (query) against the modeled data source (multidimensional data model).]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Hall for the purpose of applying a better and efficient graphical user interface for the display of Hall.

Claim 6 :

Hall discloses:

"the relational query specifies a detail filter against the relational model, and wherein the constructed multidimensional query specifies that the detail filter be applied to the modeled multidimensional data source" [abstract].

However, Hall does not explicitly disclose "the graphical user interface is further configured to enable selection of a detail filter to apply against the relational model".

On the other hand, Cras discloses, abstract, creating an analytic report on top of a multidimensional data model built on top of a relational or multidimensional database. The database operates in a computer system and provides returned values responsive to queries. Such a query is generated automatically and deduced from a report definition. 0075, disclosing that such manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.

Accordingly, Cras discloses "the graphical user interface is further configured to enable selection of a detail filter" [abstract, query is generated automatically and deduced from a report definition (selection of detail filter). 0075, dragging and dropping graphical objects]

Both Hall and Cras are directed to systems of multidimensional data, and are therefore both within the same field of endeavor. Cras discloses in more explicit detail the use of graphical objects in order to create a report and query information as described in detail above. For the above reasons it would have been obvious to one of an ordinary skill in the art to have applied Cras's disclosure of dragging and dropping the graphical objects onto the report while the user is building the report and that a query is automatically generated and deduced from a report definition to the system of Hall in order to provide a more explicit means of communicating to the system the information the user requires. As stated in Cras, 0043, the GUI enables associated

dimension objects to be filtered by selected member data so that report data is only replicated for selected members.

Claim 7 :

The combination of Hall and Cras discloses in Hall “wherein the relational query is expressed in SQL, and wherein the detail filter specified by the relational query is an SQL WHERE clause.” [col. 10 line 49]

Claim 8 :

The combination of Hall and Cras disclose in Hall “wherein the relational query specifies an aggregation function against the relational model, and wherein the constructed multidimensional query specifies that the aggregation function be applied to the modeled multidimensional data source.”[col. 11 line 1 and col. 12 lines 45-50]

Claim 9 :

The combination of Hall and Cras disclose in Hall “wherein the relational query is expressed in SQL, and wherein the aggregation function specified by the relational query is an SQL GROUP BY clause.”[col. 11 lines 1-5]

Claim 10 :

Hall discloses "the relational query specifies a summary filter against the relational model, and wherein the constructed multidimensional query specifies the summary filter be applied to the

modeled multidimensional data source" [col. 10 lines 59-62 and col. 12 lines 44-50]

However does not explicitly disclose "wherein the graphical user interface enables selection of a summary filter to apply against the relational model" On the other hand, Cras discloses, abstract, creating an analytic report on top of a multidimensional data model built on top of a relational or multidimensional database. The database operates in a computer System and provides returned values responsive to queries. Such a query is generated automatically and deduced from a report definition. 0075, disclosing that such manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.

Accordingly, Cras discloses "wherein the graphical user interface is further configured to enable selection of a summary filter to apply against the relational model" [abstract, query is generated automatically and deduced from a report definition (selection of a summary filter). 0075, dragging and dropping graphical objects]

Both Hall and Cras are directed to systems of multidimensional data, and are therefore' both within the same field of endeavor. Cras discloses in more explicit detail the use of graphical objects in order to create a report and query information as described in detail above. For the above reasons it would have been obvious to one of an ordinary skill in the art to have applied Cras's disclosure of dragging and dropping the graphical objects onto the report while the user is building the report and that a query is automatically generated and deduced from a report definition to the system of Hall in order to provide a more explicit means of communicating to the system the information the user requires. As stated in Cras, 0043, the GUI enables associated

dimension objects to be filtered by selected member data so that report data is only replicated for selected members.

Claim 14 :

The combination of Hall and Cras disclose in Hall "retrieving metadata identifying an aggregation function used for the selected measure of the multidimensional data source;" [col. 10 lines 64-67]

"determining whether the aggregation function identified by the metadata matches the selected aggregation function; and"[col. 9 lines 61-66]

"if the aggregation function identified by the metadata matches the selected aggregation function, generating a multidimensional query against the multidimensional data source that relies on the aggregation function performed in the multidimensional data source."[col. 9 lines 61-66]

Claim 15 :

The combination of Hall and Cras disclose in Hall

"receiving, in response to submitting the multidimensional database query, a multidimensional database query result; and"[col. 13 lines 25-45]

"using a relational-to-multidimensional mapping contained by the model together with relational/multidimensional equivalency logic to construct a relational database query result based on the received multidimensional database query result."[col. 3 lines 62-67 and col. 7 lines

1-4]

Claim 16 :

The combination of Hall and Cras disclose in Hall

"determining that the received relational database query is drawn against both the relational model of the multidimensional data source and one or more native relational tables; and" [col. 3 lines 62-67 and col. 7 lines 1-4, determining that the received relational database query (sql) is drawn against both the relational model of the multidimensional data source (reference/fact tables) and one or more native relational tables (virtual tables)]

"constructing a native relational database query based on aspects of the received relational database query drawn against conventional relational tables; and"[Abstract and col. 3 lines 62-67 and col. 7 lines 1-4, constructing a native relational database query (user generates a query) based on aspects of the received relational database query (sql) drawn against conventional relational tables (virtual tables)]

"submitting the native relational database query for execution against the conventional relational tables, and wherein the constructed multidimensional database query is based on aspects of the received relational database query drawn against the relational model of the multidimensional data source, the method further comprising: receiving, in response to submitting the native relational database query,"[col. 3 lines 62-67 and col. 7 lines 1-4. submitting the native relational database query (sql) for execution against the conventional relational tables (virtual tables), and wherein the constructed multidimensional database query (converted query) is based on aspects of the received relational database query (sql) drawn

against the relational model of the multidimensional data source (fact/reference table), the method further comprising: receiving, in response to submitting the native relational database query (sql modificaiton)]

"a native relational database query result; and"[abstract, modified query]

"combining the constructed relational database query result with the received native relational database query result in accordance with the received relational database query."[col. 3 lines 62-67 and col. 7 lines 1-4. combining the constructed relational database query result with the received native relational database query result (table 5) in accordance with the received relational database query (sql)]

Claim 17 :

The combination of Hall and Cras disclose in Hall "making information about the model available for use in building the received relational database query."[abstract]

Claim 18 :

The combination of Hall and Cras disclose in Hall

"determining that the received relational database query is drawn against both the relational model of the multidimensional data source and one or more native relational tables; and"[col. 3 lines 62-67 and col. 7 lines 1-4, determining that the received relational database query (sql) is drawn against both the relational model of the multidimensional data source (reference/fact tables) and one or more native relational tables (virtual tables)]

“constructing a native relational database query based on aspects of the received relational database query drawn against conventional relational tables; and” [Abstract and col. 3 lines 62-67 and col. 7 lines 1-4, constructing a native relational database query (user generates a query) based on aspects of the received relational database query (sql) drawn against conventional relational tables (virtual tables)]

“submitting the native relational database query for execution against the conventional relational tables, and”[col. 3 lines 62-67 and col. 7 lines 1-4.]

“wherein the constructed multidimensional database query is based on aspects of the received relational database query drawn against the relational model of the multidimensional data source.”[col. 3 lines 62-67 and col. 7 lines 1-4.]

Claim 19:

Hall discloses the following claimed limitations

"a first set of instructions, executable on a processor, configured to generate a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data source,“ [abstract lines 1-2, arranged in data tables, e.g. fact/reference tables. Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Accordingly, a first set of instructions, executable on a processor, configured to generate a relational model of a multidimensional data source (fact/reference table) using one or more of a schema for the multidimensional data source (schema) and metadata for the multidimensional data source (figure 2)]

"wherein the relational model comprises a relational-to-multidimensional mapping between a virtual relational table corresponding to the multidimensional data source and the multidimensional data source, and" [Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Col. 6 lines 62-66, builds SQL based on this schema. Accordingly, wherein the relational model (fact/reference table) comprises a relational-to-multidimensional mapping (schema) between a virtual relational table (virtual table) corresponding to the multidimensional data source (data warehouse) and the multidimensional data source (data warehouse)]

"the schema and metadata are accessed from the multidimensional data source;"[Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Col. 6 lines 62-66, builds SQL based on this schema. Figure 2. Accordingly, the schema (schema) and metadata are accessed from the multidimensional data source (figure 2)]

"a second set of instructions, executable on a processor, configured to form the relational database query against the relational model of the multidimensional data source"[col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. Accordingly, a second set of instructions, executable on a processor, configured to form the relational database query (SQL) against the relational model of the multidimensional data source (fact/reference tables)]

“a third set of instructions, executable on the processor, configured to receive the relational database query, the received relational database query being drawn against the relational model of the multidimensional data source;”[col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. Accordingly, a third set of instructions, executable on the processor, configured to receive the relational database query (SQL), the received relational database query being drawn against the relational model of the multidimensional data source (fact/reference tables).]

“a fourth set of instructions, executable on the processor configured to use the relational-to-multidimensional mapping to translate the received relational database query into a multidimensional database query;”[Col. 6 lines 62-66, builds SQL based on this schema. Col. 9 lines 61-64, converts the query from one which references virtual tables and columns into one which matches a physical instance of data in the data warehouse. Accordingly, a fourth set of instructions, executable on the processor configured to use the relational-to-multidimensional mapping (schema) to translate (convert) the received relational database query (SQL) into a multidimensional database query (converted query)]

“a fifth set of instructions, executable on the processor, configured to submit the multidimensional database query for execution against the modeled multidimensional data source; and”[abstract, modified query to obtain the particular information from the database warehouse. Col. 7 lines 2-4, converts the query to utilize the most appropriate fact and reference tables and issues the modified query to database server. Accordingly,). a fifth set of

instructions, executable on the processor, configured to submit the multidimensional database query (modified query) for execution against the modeled multidimensional data source (fact/reference table).]

Hall further discloses the virtual relational table corresponding to the multidimensional data source [col. 3 lines 58-60, virtual table].

Hall does not explicitly disclose “using a graphical user interface, wherein the graphical user interface

displays a presentation layer representation of the” information " corresponding to the multidimensional data source on display”

“enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;”

“a sixth set of instructions, executable on the processor, configured to display a result of the multidimensional database query against the modeled multidimensional data source.”

On the other hand, Cras discloses “ using a graphical user interface, wherein the graphical interface [0075, manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.]

"displays a presentation layer representation of the "information "corresponding to the multidimensional data source on the display, and"[0075, manipulation can be dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source on display)]

"enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"[abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data source). Abstract a query is generated automatically and deduced from a report definition (for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

"""a sixth set of instructions, executable on the processor, configured to display a result of the constructed multidimensional database query against the modeled multidimensional data source"[0012, a system uses an inventive report then query metaphor, according to the present invention to create an analytical report on top of a multidimensional data model built on top of a relational or multidimensional database. 0074, Once the report has been defined the data to populate such a report is retrieved to build the document. Accordingly, a sixth set of instructions, executable on the processor, configured to display a result (report) of the

constructed multidimensional database query (query) against the modeled data source (multidimensional data model).]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Hall for the purpose of applying a better and efficient graphical user interface for the display of Hall.

Claim 20:

The combination of Hall and Cras disclose in Hall

“a seventh set of instructions, executable on the processor, configured to receive, in response to submitting the multidimensional database query, a multidimensional database query result; and"[table 5]

“an eighth set of instructions, executable on the processor, configured to use a relational-to-multidimensional mapping to translate the received multidimensional query result into a relational database query"[Col. 6 lines 62-66, builds SQL based on this schema. Col. 9 lines 61-64, converts the query from one which references virtual tables and columns into one which matches a physical instance of data in the data warehouse. Accordingly, an eighth set of instructions, executable on the processor configured to use a relational-to-multidimensional mapping (schema) to translate (convert) the received relational database query (SQL) into a multidimensional database query (converted query)]

Claim 21:

Hall discloses the following claimed limitations

"a processor;"[figure 1]

"a display coupled to the processor;"[figure 1]

"a modeling subsystem configured to execute on the processor and further configured to generate a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data soruce, wherein the relational model comprises a relational-to-multidimensional mapping between the virtual relational table and the multidimensional data source, and"[abstract lines 1-2, arranged in data tables, e.g. fact/reference tables. Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Accordingly, a modeling subsystem configured to execute on the processor and further configured to generate a relational model of a multidimensional data source (fact/reference table) using one or more of a schema for the multidimensional data source (schema) and metadata for the multidimensional data source (figure 2)

Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Col. 6 lines 62-66, builds SQL based on this schema. Accordingly, wherein the relational model (fact/reference table) comprises a relational-to-multidimensional mapping (schema) between a virtual relational table (virtual table) corresponding to the multidimensional data source (data warehouse) and the multidimensional data source (data warehouse)]

"the schema and metadata are accessed from the multidimensional data source;"[Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Col. 6 lines 62-66, builds SQL based on this schema. Figure 2. Accordingly, the schema (schema) and metadata are accessed from the multidimensional data source (figure 2)]

"and further configured to form the relational database query against the relational model of the multidimensional data source" [col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. Accordingly, further configured to form the relational database query (SQL) against the relational model of the multidimensional data source (fact/reference tables)]

"a query reception subsystem configured to execute on the processor and further configured to receive the relational database query, the received relational database query being drawn against the relational model of the multidimensional data source;" [col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. Accordingly, a query reception subsystem configured to execute on the processor and further configured to receive the relational database query (SQL), the received relational database query being drawn against the relational model of the multidimensional data source (fact/reference tables)]

"a multidimensional query construction subsystem configured to execute on the processor and further configured to use the relational-to-multidimensional mapping to construct a multidimensional database query based on the received relational database query; and" [Col. 6

lines 62-66, builds SQL based on this schema. Col. 9 lines 61-64, converts the query from one which references virtual tables and columns into one which matches a physical instance of data in the data warehouse. Accordingly, a multidimensional query construction subsystem configured to execute on the processor and further configured to use the relational-to-multidimensional mapping (schema) to translate (convert) the received relational database query (SQL) into a multidimensional database query (converted query)]

“a query submission subsystem configured to execute on the processor and further configured to submit the constructed multidimensional database query for execution against the modeled multidimensional data source." [abstract, modified query to obtain the particular information from the database warehouse. Col. 7 lines 2-4, converts the query to utilize the most appropriate fact and reference tables and issues the modified query to database server. Accordingly, a query submission subsystem configured to execute on the processor and further configured to submit the multidimensional database query (modified query) for execution against the modeled multidimensional data source (fact/reference table).]

Hall further discloses discloses the virtual relational table corresponding to the multidimensional data source [col. 3 lines 58-60, virtual table].

Hall does not explicitly disclose:

“a graphical user interface subsystem configured to execute on the processor” “wherein the graphical user interface subsystem further

Displays a presentation layer representation of" information "corresponding to the multidimensional data source on the display,

Enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"

On the other hand, Cras discloses " a graphical user interface subsystem configured on the processor" "wherein the graphical user interface subsystem further" [0075, manipulation can be dragging and dropping the graphical objects onto the report while the user is building the report.]

> "displays a presentation layer representation of the "information "corresponding to the multidimensional data source on the display, and"[0075, manipulation can be dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source on the display)]

> "enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;"[abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data

source). Abstract a query is generated automatically and deduced from a report definition (for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Hall for the purpose of applying a better and efficient graphical user interface for the display of Hall.

Claim 22:

The combination of Hall and Cras disclose in Hall "a query result reception subsystem that receives, in response to submitting the multidimensional database query, a multidimensional database query result; and"[table 5]

"a relational query result construction subsystem that uses a relational-to-multidimensional mapping to construct a relational database query result based on the received multidimensional database query result."[Col. 9 lines 10-15, one supporting method involves the use of SQL views which map a native relational schema model to a schema model that an OLAP API can understand. An OLAP API to relational database can recognize the views and hence the type and form of data returned. Thus, an OLAP API can generate an appropriate SQL statement for querying OLAP objects. Accordingly, disclosing a relational query result construction subsystem that uses a relational-to-multidimensional mapping (map) to translate (bridge) the received relational database query (generate SQL) into a multidimensional database query (querying OLAP objects)]

Claim 31:

Hall discloses the following claimed limitations:

“generating a relational model of a multidimensional data source using one or more of a schema for the multidimensional data source and metadata for the multidimensional data source," [abstract lines 1-2, arranged in data tables, e.g. fact/reference tables. Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Accordingly, generating a relational model of a multidimensional data source (fact/reference tables) using one or more of a schema for the multidimensional data source (schema) and metadata for the multidimensional data source(figure 2),]

“wherein the relational model comprises a relational-to-multidimensional mapping between a virtual relational table corresponding ot the multidimensional data source and the multidimensional data source, and"[Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is independent of which summaries are populated. Figure 2. Col. 6 lines 62-66, builds SQL based on this schema. Accordingly, wherein the relational model comprises a relational-to-multidimensional mapping (schema) between the virtual relational table (virtual table) and the multidimensional data source (datawarehouse)]

“the schema and metadata are accessed from the multidimensional data source;” [Col. 6 lines 14-17, schema which abstracts the definition of the physical data warehouse into one which is

independent of which summaries are populated. Figure 2. Col. 6 lines 62-66, builds SQL based on this schema. Figure 2. Accordingly, the schema (schema) and metadata are accessed from the multidimensional data source (figure 2)]

“forming the relational database query against a relational model of a multidimensional data source” [col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. Accordingly, forming the relational database query (SQL) against the relational model of the multidimensional data source (fact/reference tables)]

“receiving the relational database query, the received relational database query being drawn against both the relational model of a multidimensional data source and a native relational table;"[col. 6 lines 64, builds SQL statement. Col. 7 lines 2-3, fact table and reference tables. receiving the relational database query (SQL), the received relational database query being drawn against the relational model of the multidimensional data source (fact/reference tables)]

“converting the received relational database query into (1) a native relational database query against only the native relational table, and (2) a multidimensional database query against the multidimensional data source;” [Col. 6 lines 62-66, builds SQL based on this schema. Col. 9 lines 61-64, converts the query from one which references virtual tables and columns into one which matches a physical instance of data in the data warehouse. Accordingly, converting the received relational database query (query) into (1) a native relational database query (SQL)

against only the native relational table (virtual table), and (2) a multidimensional database query (modified query) against the multidimensional data source (data warehouse)]

“submitting the native relational database query against the native relational table;"[Col. 9 lines 61-64, converts the query from one which references virtual tables and columns into one which matches a physical instance of data in the data warehouse. Accordingly, submitting the native relational database query (query) against the native relational table (virtual table)]

“submitting the multidimensional database query against the multidimensional data source;”[abstract, modified query to obtain the particular information from the database warehouse. Col. 7 lines 2-4, converts the query to utilize the most appropriate fact and reference tables and issues the modified query to database server. Accordingly, submitting the multidimensional database query (modified query) against the multidimensional data source (data warehouse)]

Hall further discloses the virtual relational table corresponding to the multidimensional data source [col. 3 lines 58-60, virtual table].

Hall does not explicitly disclose

“using a graphical user interface, wherein the graphical user interface”

“displays a presentation layer representation of the” information “corresponding to the multidimensional data source, and”

“enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;”

“combining contents of a first search result produced in response to the native relational database query and a second search result produced in response to the multidimensional database query into a third search result responsive to the received relational database query; and”

“displaying the third search result.”

On the other hand, Cras discloses

“using a graphical user interface, wherein the graphical user interface”[0075, gui]

“displays a presentation layer representation of the” information “corresponding to the multidimensional data source, and”[[0075, manipulation can be dragging and dropping the graphical objects (gui interface displays a presentation layer) onto the report while the user is building the report (representation of information corresponding to the multidimensional data source)]]

“enables pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation layer;”

[abstract, built on top of a relational or multidimensional database (one or more tables and columns of data stored in multidimensional data source). Abstract a query is generated automatically and deduced from a report (and represented by the displayed presentation layer) definition (for database query of). 0075, dragging and dropping (enable pointer-driven selection) the graphical objects (represented graphical objects).]

"combining contents of a first search result produced in response to the native relational database query and a second search result produced in response to the multidimensional database query into a third search result responsive to the received relational database query; and"[0089, report engine creates a datasource query consisting of a tree of blockqueries and provides the dsquery to the cube. 0091, a datasource query is used to access and retrieve data from the OLAP and RDBMS data sources. 0104-0105. combining contents (report) of a first search result produced in response to the native relational database query (relational query) and a second search result produced in response to the multidimensional database query (multidimensional query) into a third search result (report) responsive to the received relational database query (relational query)]

"displaying the third search result."[0132, datasource query contains the information necessary to build a query of the report, then query process report]

It would have been obvious to a person of an ordinary skill in the art at the time the invention was made to have applied the disclosure of Cras above to the disclosure of Hall for the purpose of applying a better and efficient graphical user interface for the display of Hall.

10. Claims 4-5 and 12-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. patent 5675785 by Hall et. al. (hereafter Hall) and U.S. Patent application publication 20020091681 by Cras et. al. (hereafter Cras) further in view of U.S. patent application publication 20040039736 by Kilmer et. al. (hereafter Kilmer).

Claim 4 :

Hall and Cras do not explicitly disclose "wherein the multidimensional query is constructed in MDX."

However, Kilmer discloses 0019, "multi-dimensional expression query statement mdx". It would have been obvious to one of an ordinary skill in the art to apply KIimer's disclosure of MDX to the combination of Hall and Cras for the purpose of being able to talk to the multidimensional data source in a language it would understand. It is further well known in the art that MDX is a standard multidimensional database query language.

Claim 5 :

The combination of Hall, Cras, and Kilmer disclose in Hall “wherein the relational query is expressed in SQL.” [col. 6 line 64, sql]

Claim 12 :

The combination of Hall and Cras disclose the following limitations:

Hall discloses "the relational query specifies the detail filter against the relational model having selected predicates, and" [col. 10 lines 45-50]

Cras discloses "the graphical user interface is further configured to enable selection of a detail filter to apply against the relational model, and" [abstract, query is generated automatically and deduced from a report definition (selection of a summary filter). 0075, dragging and dropping graphical objects]

However Cras and Hall do not explicitly disclose "the constructed multidimensional query specifies, for each of the selected predicates that can be applied against the modeled multidimensional data source before a crossjoin operation is performed, applying the selected predicate against the modeled multidimensional data source as early as possible".

On the other hand, Kilmer discloses, "the constructed multidimensional query specifies, for each of the selected predicates that can be applied against the modeled multidimensional data source before a crossjoin operation is performed, applying the selected predicate against the modeled multidimensional data source as early as possible" [0073, the mdx cross join function is used to join the member and expression strings from each dimension strings together (e.g. predicates that can be applied against the modeled multidimensional data source before crossjoin). Then once the member and expression strings have been established for all of the axes, a select clause is built by concatenating the columns and rows strings together.]. It would have been obvious to one of an ordinary skill in the art at the time the invention was made to have applied Kilmer's disclosure of the above for the purpose of processing an MDX query.

Claim 13 :

The combination of Hall and Cras disclose the following limitations:

Hall discloses "the relational query specifies the detail filter against the relational model having selected predicates, and" [col. 10 lines 46-49]

Cras discloses "the graphical user interface is further configured to enable selection of a detail filter to apply against the relational model, and" [abstract, query is generated automatically and deduced from a report definition (selection of a summary filter). 0075, dragging and dropping graphical objects]

However Cras and Hall do not explicitly disclose "the constructed multidimensional query specifies, for each of the selected predicates that can be applied against the modeled multidimensional data source before a crossjoin operation is performed, applying the selected predicate against the modeled multidimensional data source before the crossjoin operation is performed."

On the other hand, Kilmer discloses, "the constructed multidimensional query specifies, for each of the selected predicates that can be applied against the modeled multidimensional data source before a crossjoin operation is performed, applying the selected predicate against the modeled multidimensional data source as early as possible" [0073, the mdx cross join function is used to join the member and expression strings from each dimension strings together (e.g. predicates that can be applied against the modeled multidimensional data source before crossjoin). Then once

the member and expression strings have been established for all of the axes, a select clause is built by concatenating the columns and rows strings together.].

Hall, Cras, and Kilmer are all within the same field of endeavor, namely multidimensional systems. It would have been obvious to one of an ordinary skill in the art to have applied Kilmer's disclosure of the above for the purpose of processing an MDX query.

11. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. patent 5675785 by Hall et. al. (hereafter Hall) and U.S. Patent application publication 20020091681 by Cras et. al. (hereafter Cras) further in view of "Database Systems: the complete book" by Hector Garcia-Molina et. al. (hereafter Garcia-Molina).

<u>Claim 11 :</u>

The combination of Hall and Cras discloses in Hall "wherein the relational query is expressed in SQL" [col. 10 lines 45-50], "and wherein the summary filter specified by the relational query" [col. 10 lines 59-62].

However, Hall does not explicitly disclose is an "SQL HAVING clause".

On the other hand, Garcia-Molina discloses that SQL supports grouping, by using a GROUP BY clause [pages 277, 282]. All references are in the same field of endeavor, databases. It would have been obvious to have applied Garcia-Molina's disclosure above to the combination of Hall and Cras for the purpose of grouping tuples together. One of ordinary skill in the art would know this would be a form of filtering.

Response to Arguments

12. Applicant's arguments with respect to claims 3-22 and 31 have been considered but are moot in view of the new ground(s) of rejection.

Applicant’s assert the following directed towards the Cras reference:

A. Cras does not provide disclosure of the graphical interface, but5 merely to a purported normalized model, page 18 of remarks lines 27-28. That Cras does not involve a graphical user interface because there is no indication from the cited sections that it permits "pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation.” That the cited sections do not provide for drag and drop, and that Applicant’s respectfully submit that Cras fails to provide any disclosure of "drag and drop" or even a graphical user interface at all.

In response, the examiner respectfully disagrees. Cras discloses a graphical user interface, 0075 line 3. Therefore, applicant’s asserted no graphical user interface at all is unpersuasive. 0075 lines 2-7 states that a user manipulates graphical objects as presented to a user at a graphical user interface (GUI), where the objects represent the available data at the data source. Further disclosing, 0075, such manipulation can be by dragging and dropping the graphical objects onto the report whil the user is building the report. Furthermore, 0084, states

that the user interface of the client provides a user with the ability to build a report by graphically dragging and dropping the report objects. Therefore, applicant's asserted no disclosure of drag and drop is unpersuasive.

In regards to pointer-driven selection for database query of one or more tables and columns of data stored in the multidimensional data source and represented by the displayed presentation. Cras discloses 0075 the system and method of the present invention will generate the relevant query depending on where in the report an object is dropped. 0089, if specific datasource is accessing multidimensional data source of 212 of figure 2, then specific datasource 316 accesses data from multidimensional servers using the OLAP data access model. In other words, Cras accordingly, discloses pointer-driven selection (drag and drop) for database query (query) of one or more tables and columns of data (data) stored in the multidimensional data source (multidimensional source) and represented by the displayed presentation (report). Therefore, applicant's assertions in regards to Cras are unpersuasive.

Conclusion

13. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after

the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Contact Information

14. Any inquiry concerning this communication or earlier communications from the examiner should be directed to MICHAEL PHAM whose telephone number is (571)272-3924. The examiner can normally be reached on 9am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Cottingham can be reached on 571-272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/M. P./
Examiner, Art Unit 2167

/Luke S. Wassum/
Primary Examiner
Art Unit 2167